

Sparse Named Entity Classification using Factorization Machines

Ai Hirata and Mamoru Komachi

Tokyo Metropolitan University

6-6 Asahigaoka, Hino, Tokyo 191-0065, Japan

hirata-ai@ed.tmu.ac.jp and komachi@tmu.ac.jp

Abstract

Named entity classification is the task of classifying text-based elements into various categories, including places, names, dates, times, and monetary values. A bottleneck in named entity classification, however, is the data problem of sparseness, because new named entities continually emerge, making it rather difficult to maintain a dictionary for named entity classification. Thus, in this paper, we address the problem of named entity classification using matrix factorization to overcome the problem of feature sparsity. Experimental results show that our proposed model, with fewer features and a smaller size, achieves competitive accuracy to state-of-the-art models.

to consider the relationships between sparse features. Through our experiments, we achieved competitive accuracy to models developed in previous works in terms of using fewer features and compactness using factorization machines (Rendle, 2010). The main contributions of this paper are as follows:

- We address the data sparseness problem in unknown named entity classification using factorization machines.
- We demonstrate that factorization machines achieve state-of-the-art performance in sparse named entity classification task using a reduced feature set and a compact model.

1 Introduction

To date, standard approaches to named entity classification rely on supervised models, that typically require a large-scale annotated corpus and a wide-coverage dictionary. However, since new named entities arise regularly, it becomes increasingly difficult to maintain an up-to-date dictionary and/or adapt a named entity classifier to a new domain; for example, sequence labeling techniques that use feature templates (Finkel et al., 2005; Sarawagi and Cohen, 2004) are not robust for unknown named entities because their feature space is very sparse (Primadhanty et al., 2015). This problem worsens when we attempt to use a combination of features for sparse named entity classification.

Therefore, in this paper, we propose the use of matrix factorization for named entity classification

2 Related Work

A standard approach to named entity classification is to formulate a task as a sequence labeling problem and use a supervised method, such as conditional random fields (Lafferty et al., 2001; Finkel et al., 2005; Sarawagi and Cohen, 2004). These studies heavily rely on feature templates for learning combinations of features; however, since combinations of features in conventional supervised learning are treated independently, this approach is not robust for named entities that do not appear in the training data.

To address the task of unknown named entity classification, Primadhanty et al. (2015) explored the use of sparse combinatorial features. They proposed a log-bilinear model that defines a score function considering interactions between features; the score function is regularized via a nuclear norm on a feature weight matrix. Further, their method employs

singular value decomposition (SVD)-based regularization to handle the combination of features. They reported that their regularization achieved higher accuracy than L1 and L2 regularization, frequently used in natural language processing (Okanohara and Tsujii, 2009).

However, nuclear norm regularization (i.e., SVD-based regularization) is not necessarily the best way to incorporate interactions between features, because it does not directly optimize classification accuracy. Therefore, our proposed method treats sparse features using matrix factorization from a different perspective: we decompose a feature weight matrix using factorization machines as to directly optimize classification accuracy using a large margin method similar to support vector machines (SVMs) and passive-aggressive algorithms (Vapnik, 1995; Crammer et al., 2006).

3 Factorization Machines

In this paper, we propose the use of factorization machines (Rendle, 2010) for unknown named entity classification. Using this approach, we can employ the same objective function as SVMs and yet performs matrix factorization to handle sparse combinatorial features. Matrix factorization yields better generalizations over a sparse feature matrix (Madhyastha et al., 2014).

Factorization machines with interaction degree $d = 2$ use the following equation for prediction:

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n \langle v_i, v_j \rangle x_i x_j \quad (1)$$

Here, \mathbf{x} is an instance, x_i represents the i -th dimension of the feature x , n is the number of features, $w \in \mathbb{R}^n$ is a weight vector, and $w_0 \in \mathbb{R}$ is a bias term. Factorization machines incorporate interactions between variables v_i, v_j as the third term of Equation (1). Here, $\langle \cdot, \cdot \rangle$ is the inner product of two vectors of size k , i.e.,

$$\langle v_i, v_j \rangle := \sum_{f=1}^k v_{i,f} \cdot v_{j,f} \quad (2)$$

where v_i is the i -th element of matrix $\mathbf{V} \in \mathbb{R}^{n \times k}$ and k is a hyperparameter representing the dimension of matrix decomposition. To consider interactions between features, we only need to calculate the inner

	training		development		test
PER	6,516	(3,489)	1,040	(762)	1,342 (925)
LOC	6,159	(987)	176	(128)	246 (160)
ORG	5,721	(2,149)	400	(273)	638 (358)
MISC	3,205	(760)	177	(142)	213 (152)
O	36,673	(5,821)	951	(671)	995 (675)

Table 1: Number of candidates (i.e., tokens) in the dataset obtained from Primadhanty et al. (2015), with the number of unique candidates (i.e., types) shown in parentheses.

	P	R	F1
log-linear model	49.75	44.50	46.75
SVM (polynomial kernel)	53.75	50.67	51.94
log-bilinear model (Primadhanty et al., 2015)	62.03	53.92	55.88
factorization machines	60.93	55.10	57.27

Table 2: Results of unknown named entity classification.

product of a decomposed matrix $n \times k$ times. Therefore, we do not incur high computational costs even though the number of interacting features is large.

Note that even though the polynomial kernel of SVMs take combinations of features into account, it treats them independently. Conversely, factorization machines take advantage of interactions between features using a low-dimensional feature matrix via matrix factorization. Because factorization machines can learn combinations of infrequent features thanks to matrix factorization, we expect that factorization machines will correctly classify named entities that seldom appear in the training corpus.

In a binary classification task, factorization machines use hinge loss to optimize parameters. Here, parameter learning can be accomplished via Markov chain Monte Carlo or stochastic gradient descent.

4 Experiments

As described above, we aim to classify named entities that rarely appear in a given training corpus. We compared factorization machines with a log-linear model, a polynomial-kernel SVM, and a state-of-the-art log-bilinear model using nuclear norm for regularization (Primadhanty et al., 2015).

4.1 Settings

Data. We used the dataset provided by Primadhanty et al. (2015); this dataset was created for evaluating unknown named entity classification and is

context features: Right and left contexts of the candidate in a sentence (do not take the order into account).
cap=1, cap=0: Whether the first letter of the candidate is uppercase, or not.
all-low=1, all-low=0: Whether all letters of the candidate are lowercase, or not.
all-cap1=1, all-cap1=0: Whether all letters of the candidate are uppercase, or not.
all-cap2=1, all-cap2=0: Whether all letters of the candidate are uppercase and periods, or not.
num-tokens=1, num-tokens=2, num-tokens>2: Whether the candidate consists of 1, 2, or more tokens.
dummy: Dummy feature to capture context features.

Table 3: Features used in our experiment; note that this is a subset of features used in Primadhanty et al. (2015)’s experiment.

based on the CoNLL-2003 English dataset, which omits named entity candidates that appear in the training data from the development and test data.

Table 1 shows the number of tokens and types in the given dataset. This dataset contains five tags: person (PER), location (LOC), organization (ORG), miscellaneous (MISC), and non-entities (O).

Features. We used a subset of features from experiments performed by Primadhanty et al. (2015). Table 3 summarizes the features used in our experiment, including context and entity features.

Tools. In terms of tools, we used scikit-learn 0.17 to implement a log-linear model and polynomial kernel in an SVM. Further, we employed libFM 1.4.2¹ (Rendle, 2012) to build a named entity classifier using factorization machines.

In the interaction of both the SVM and the factorization machine, we fixed the degree of the polynomial kernel to $d = 2$. We also tuned other parameters such as learning methods, learning rate and regularization methods based on development data. Further, we used a one-versus-all strategy to build a multiclass classifier.

Evaluation metrics. For our evaluation, we used precision, recall, and F1-score. The scores were calculated on all tags except for non-entities (O).

4.2 Results

Table 2 presents results of our experiments. Note that Primadhanty et al. (2015) used additional features such as Brown clustering and parts-of-speech (POS) features, which we did not use. Table 4 and Figure 2 show the performance and precision-recall curves of named entity classification for each tag, respectively.

¹<http://www.libfm.org/>

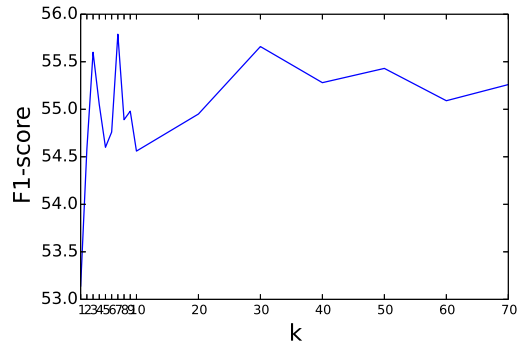


Figure 1: F1-score on development data as varying dimension k varies using factorization machines.

We observed here that, aside from LOC, we obtained competitive results to the state-of-the-art named entity classifier proposed by Primadhanty et al. (2015) with fewer features. Overall, the micro-averaged F1 score improved by 1.4 points.

From these results, we conclude that unknown named entity classification can be successfully achieved by taking combinatorial features into account using factorization machines.

5 Discussion

Experimental results show that performance on ORG was improved. For example, the term “Vice-President” appears in both contexts of ORG and O, and our method correctly handled this sparse combination of context and entity features.

The accuracy of LOC, however, was lower than that of the log-bilinear model (Primadhanty et al., 2015). Upon investigating the confusion matrix, we found that the LOC tag was often misclassified as PER. We therefore conclude here that clustering and POS features are necessary to distinguish these tags.

Figure 1 plots the F1-score of our proposed method as dimension k changes for matrix factor-

	PERSON			LOCATION			ORGANIZATION			MISC		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
SVM (polynomial kernel)	86.45	72.28	78.73	31.35	38.62	34.61	62.54	59.40	60.93	34.67	32.39	33.50
log-bilinear model (Primadhanty et al., 2015)	73.83	90.84	81.46	64.96	36.18	46.48	72.11	44.98	55.41	37.20	43.66	40.17
factorization machines	84.36	80.40	82.33	39.49	50.41	44.29	70.88	55.33	62.15	48.99	34.27	40.33

Table 4: A breakdown of the results of unknown named entity classification per tag.

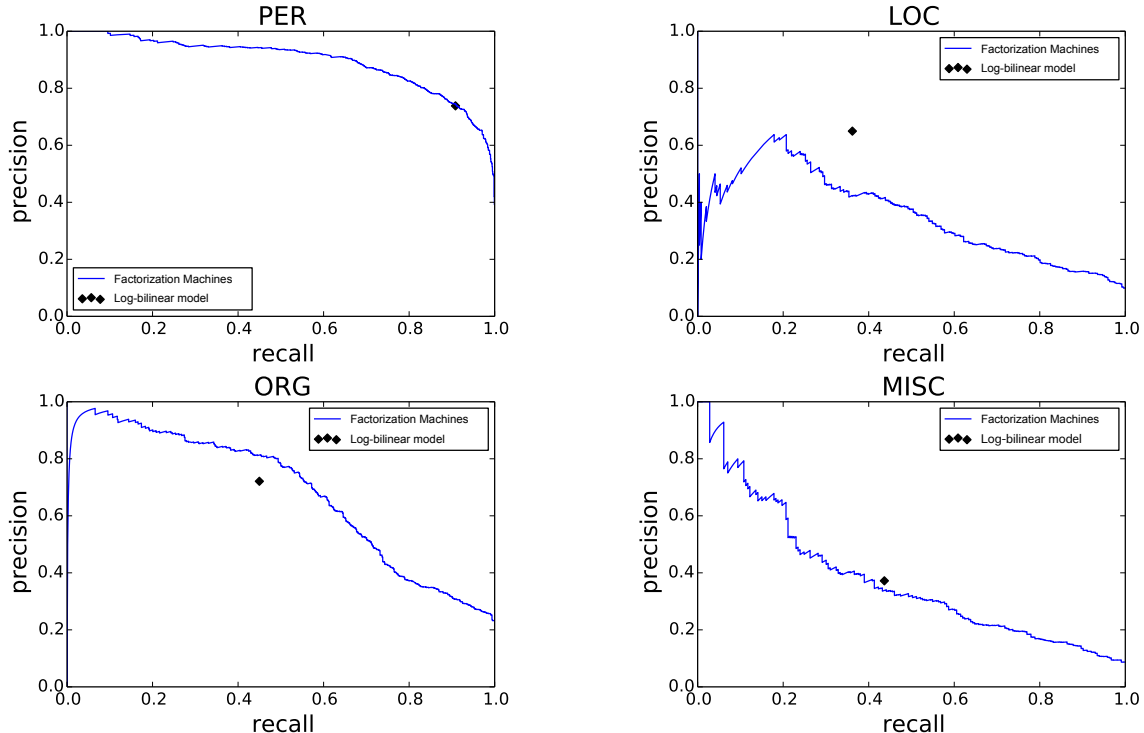


Figure 2: Precision-recall curves of unknown named entity classification for each tag.

ization using the same development data as that of Primadhanty et al. (2015). Our method yielded the best F1-score (i.e., 57.1) at $k = 5$, whereas the log-bilinear model achieved the best F1-score (i.e., 61.73) at $k = 40$. These results show that factorization machines require a compact model to achieve state-of-the-art results on the test set of this corpus.

It would be interesting to point out that the performance of our factorization machines approach on the development dataset was lower than that of the log-bilinear model by 4.6 points. This phenomenon may occur because the log-bilinear model overfits to sparse combinatorial features even with nuclear norm regularization; further factorization machines typically have better generalization abilities than those of nuclear norm regularization.

Both our approach and the methods of Primadhanty et al. (2015) address the problem of incor-

porating sparse combinatorial features by dimension reduction (i.e., matrix factorization); however, they differ in terms of the objective function to be optimized. Primadhanty et al. (2015) use maximum likelihood estimation as an objective function; whereas other objective functions such as hinge loss can be used in factorization machines.

6 Conclusion

In this paper, we proposed the use of factorization machines to handle the combinations of sparse features in unknown named entity classification. Our experimental results showed that we were able to achieve competitive accuracy to state-of-the-art methods using fewer features and a compact model. For future work, we aim to extend this framework to sequence labeling, thereby improving overall named entity recognition.

References

- [Crammer et al.2006] Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. 2006. Online Passive-Agressive Algorithms. *The Journal of Machine Learning Research*, 7:551–585.
- [Finkel et al.2005] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of ACL*, pages 363–370.
- [Lafferty et al.2001] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of ICML*, pages 282–289.
- [Madhyastha et al.2014] Pranava Swaroop Madhyastha, Xavier Carreras, and Ariadna Quattoni. 2014. Learning Task-specific Bilexical Embeddings. In *Proceedings of COLING*, pages 161–171.
- [Okanohara and Tsujii2009] Daisuke Okanohara and Jun’ichi Tsujii. 2009. Learning Combination Features with L_1 Regularization. In *Proceedings of NAACL*, pages 97–100.
- [Primadhanty et al.2015] Audi Primadhanty, Xavier Carreras, and Ariadna Quattoni. 2015. Low-Rank Regularization for Sparse Conjunctive Feature Spaces: An Application to Named Entity Classification. In *Proceedings of ACL-IJCNLP*, pages 126–135.
- [Rendle2010] Steffen Rendle. 2010. Factorization Machines. In *Proceedings of ICDM*, pages 995–1000.
- [Rendle2012] Steffen Rendle. 2012. Factorization Machines with libFM. *ACM Transactions on Intelligent Systems and Technology*, 3(3):57:1–57:22.
- [Sarawagi and Cohen2004] Sunita Sarawagi and William W. Cohen. 2004. Semi-Markov Conditional Random Fields for Information Extraction. In *Proceedings of NIPS*, pages 1185–1192.
- [Vapnik1995] Vladimir N. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer.